

## **REMARKS**

In response to the Office Action mailed December 5, 2007, Applicant respectfully requests reconsideration. Claims 1-28 were previously pending in this application. By this amendment, claims 1 and 15 have been amended. As a result, claims 1-28 are pending for examination with claims 1, 8, 15, and 22 being independent. No new matter has been added.

### **Objections to the Specification**

The Office Action objected to the specification and advised the Applicant to amend paragraph [0004] of the specification to cancel the changes required by previously made objection to the specification. Applicant has amended the specification to address the objection.

Accordingly, withdrawal of this objection is respectfully requested.

### **Rejections Under 35 U.S.C. §103 I**

The Office Action rejected claims 1, 4-7, 15, 18-21 under 35 U.S.C. 103(a) as allegedly being unpatentable over Boyd et al., U.S. Patent No. 6,721,806 ("Boyd") in view of Applicant's admitted prior art. Applicant respectfully disagrees. In addition, without acceding to the appropriateness of the rejections, Applicant has amended independent claims 1 and 15 to more clearly distinguish over the cited reference.

#### **A. Independent Claim 1**

Claim 1, as amended, recites:

A method for transferring control between a first network interface and at least a second network interface in a multiple network interface device, after the first network interface sends an identifier, the identifier associated with a memory location in the multiple network interface device, to a second device, the identifier and an associated data field capable of being received by the second network interface in the multiple network interface device, the method comprising:

receiving a message from the second network interface in the multiple network interface device by a program component of the multiple network interface device, the message indicating the reception of the identifier and the

associated data field from the second device, wherein the second network interface has no knowledge of the identifier and the associated data field;  
passing the identifier to the program component;  
querying the first network interface to supply the program component with a list of identifiers generated by the first network interface and associated memory locations in multiple network interface device memory;  
identifying, by the program component, that the first network interface generated the identifier; and  
transmitting a memory location associated with the identifier to the second network interface, the second network interface capable of transmitting the associated data field to the memory location associated with the identifier.

On pages 4-5, the Office Action states that Boyd shows a method for transferring control “between a first network interface and at least a second network interface [IPSOE (118) and (120)]” in a multiple network interface device, after the first network interface sends an identifier [a send work queue (402) wherein each of the send work request’s data segments contains part of a virtually contiguous memory region (col. 8, lines 38-48) specified by DDP/RDMA header (622)], the identifier associated with a memory location in the multiple network interface device [DDP/RDMA header (622) that specifies the message identifier and the placement information for the data payload (col. 11, lines 10-14)], to a second device, the identifier and an associated data field capable of being received by the second network interface (col. 11, lines 36-55).”

Boyd describes that DDP/RDMA header 622 specifies **the message identifier** and the placement information for the data payload. (Boyd, col. 11, lines 10-11). The message identifier is constant for all frames that are part of a message. (Boyd, col. 11, lines 11-13). **Example message identifiers include**, for example, **send**, **write RDMA**, and **read RDMA**. (Boyd, col. 11, lines 13-14). (Emphasis added). Thus, the Office Action apparently assumes that a first network interface and a second network interface are IPSOE 118 and IPSOE 120, respectively. Further, the identifier referred to in the Office Action include, for example, send, write RDMA and read RDMA. Since the Office Action refers to a send work queue as disclosing “the first network interface sends an identifier,” it may be reasonable to assume that the Office Action implies that a message identifier associated with each of the send work request’s data is, for example, send, write RDMA and read RDMA.

Claim 1 recites that the identifier and an associated data field capable of being received by the second network interface in the multiple network interface device. Boyd discusses that

distributed computer system 700 in FIG. 7 includes a host processor node 702 running process A 716 and a host processor node 704 running process B 718. (Boyd, col. 11, lines 17-20; Fig. 7) In col. 11, lines 36-55, Boyd discusses that process A initiates a message request by posting work queue elements to the send queue of queue pair 23. (Boyd, col. 11, lines 36-37). Hardware in host IPSOE 706 reads the work queue element and segments the message stored in virtual contiguous buffers into data frames, such as the data frame illustrated in FIG. 6. (Boyd, col. 11, lines 47-50). Data frames are routed through the IP net fabric, and for reliable transfer services, are acknowledged by the **final destination endnode**. (Boyd, col. 11, lines 50-52). (Emphasis added). Thus, Boyd describes that the work queue elements routed to the final destination endnode. In contrast, claim 1 recites that the identifier and an associated data field capable of being received by *the second network interface in the multiple network interface device*. (Emphasis added).

Further, on page 5, the Office Action alleges that Boys shows receiving a message from the second network interface by a program component, the message indicating the reception of the identifier and the associated data field from the second device as “receive work queue (400) containing work queue elements (WQEs) (416-420), describing where to place incoming channel semantic data from the IP net fabric (col. 8, lines 16-22) wherein a program component comprises a message and data service (224) that receives and processes messages from the second network interface comprising IPSOE (212) (col. 7, lines 11-23).” In col. 8, lines 16-22, Boyd describes the following: “[s]end work queue 402 contains work queue elements (WQES) 422 - 428, describing data to be transmitted on the IP net fabric. Receive work queue 400 contains work queue elements (WQEs) 416 - 420, describing where to place incoming channel semantic data from the IP net fabric. A work queue element is processed by hardware 408 in the IPSOE.” Boyd does not discuss the message indicating the reception of the identifier and the associated data field *from the second device*. (Emphasis added). Boyd discusses that host processor node 200 includes a message and data service 224, which is a higher-level interface than the verb layer and is used to process messages and data received **through IPSOE 210 and IPSOE 212**. (Boyd, col. 7, lines 17-21). (Emphasis added). Thus, while the Office Action states that “Boyd shows ... [that] a program component comprises a message and data service (224) that receives and processes messages from the second network interface comprising IPSOE (212),” Boyd discusses that a message and data service 224 is used to process messages and data received through **both** IPSOE 210 and IPSOE 212. (Emphasis added).

Furthermore, the Office Action alleges that Boys shows “querying the first network interface to supply the program component with a list of identifiers [the range of QP numbers, CQ numbers, and the range of Memory Translation and Protection Table (col. 21, lines 20-24) which provide a list of identifiers and associated memory locations (col. 7, lines 39-44; col. 16, lines 1-40)] sent by the first network interface and associated memory locations in multiple network interface device memory [(col. 21, lines 6-12) wherein the host comprises the program components (col. 7, lines 17-24)].” As discussed above, while rejecting the first limitation of claim 1, the Office Action stated that the first network interface is shown as IPSOE 118 in Boyd. However, in col. 21, lines 20-24 Boyd discusses that [a]s shown in FIG. 13B, the [resynchronization] operation starts with **a query of the Primary RNIC to determine the range of QP numbers, the range of CQ numbers, and the range of Memory Translation and Protection Table entries** that will be configured for S/S and non-S/S (step 1320). (Emphasis added). Thus, the Office Action refers to the primary RNIC as the first network interface. As discussed above, the Office Action states that the identifier includes send, write RDMA and read RDMA. Thus, it is not clear to which identifiers and which list of identifiers the Office Action refers to while citing to col. 21, lines 20-24 of Boyd. Moreover, Boyd does not discuss querying the first network interface **to supply the program component** with a list of identifiers, as recited in claim 1. (Emphasis added).

Subsequently, on page 6, the Office Action alleges that Boyd discloses “identifying, by the program component, that the first network interface sent the identifier” in “having RNIC identification context information (1240) as part of QP (1172) that uniquely identifies RNIC, which sent/received a message comprising the identifier (col. 16, lines 24-28 and col. 19, lines 1-10).” Claim 1 recites identifying, by the program component, that the first network interface *generated* the identifier. (Emphasis added). In col. 16, lines 24-28, Boyd discusses “[t]he traditional QP Context State 1220, contains well-known QP Context State information, such as the state of the QP, the type of service associated with the QP, TCP source port and destination numbers associated with the QP, IP source and destination addresses associated with the QP.” In col. 19, lines 1-10, Boyd states that the Primary and Alternate RNIC Identification Context 1240 contains the following information: RNIC Switchover/Switchback State, Primary RNIC Identifier, Primary RNIC Port Identifier, Alternate RNIC Identifier, Alternate RNIC Port Identifier. Thus, in these portions Boyd does not teach identifying, *by the program component*, that the first network interface generated the identifier. (Emphasis added). Moreover, Boyd states that the Primary and

Alternate RNIC Identification Context 1240 contains both Primary RNIC Identifier and Alternate RNIC Port Identifier. Thus, Boyd does not appear to describe identifying, by the program component, *that the first network interface generated the identifier*. (Emphasis added). In addition, the Office Action points out to “RNIC, which **sent/received** a message comprising the identifier,” as described above. (Emphasis added). However, claim 1 recites that *the first network interface sends an identifier ... [and] receiving a message from the second network interface* in the multiple network interface device by a program component of the multiple network interface device, the message indicating *the reception of the identifier and the associated data field* from the second device. (Emphasis added).

On page 6, the Office Action states that Boyd teaches transmitting a memory location associated with the identifier to the second network interface, the second network interface capable of transmitting the associated data field to the memory location associated with the identifier. Again, Boyd does not disclose transmitting a memory location *associated with the identifier* to the second network interface, the second network interface capable of transmitting the associated data field to the memory location *associated with the identifier*. (Emphasis added).

On page 6, the Office Action concedes that Boyd “does not show that the second network interface has no knowledge of the identifier and the associated data field, and that the first network interface generated the identifier.” The Office Action goes on to allege that Applicant’s admitted prior art shows this limitation. However, Boyd describes **sharing** memory regions between Primary RNIC and Alternate RNIC: “[f]or each Memory Region that is to be shared between the Primary RNIC and the Alternate RNIC, the host uses the Register Shared memory verb to register all shared memory regions on the Alternate RNIC (step 1326).” (Boyd, col. 21, lines 39-42). Moreover, Boyd describes the Primary and Alternate RNIC Identification Context 1240 that contains the RNIC Switchover/Switchback State, Primary RNIC Identifier, Primary RNIC Port Identifier, Alternate RNIC Identifier, Alternate RNIC Port Identifier. Therefore, Boyd appears to teach away from claim 1. Thus, a combination of Boyd and the Applicant’s admitted prior would not render claim 1 obvious.

In view of the foregoing, claim 1 patentably distinguishes over Boyd and the Applicant’s admitted prior art, either alone or in combination.

Claims 2-7 depend from claim 1 and are allowable for at least the same reasons.

Accordingly, withdrawal of the rejection of claims 1-7 is respectfully requested.

B. Independent Claim 15

Claim 15, as amended, recites:

A computer readable medium having stored therein instructions for performing acts for transferring control between a first network interface and at least a second network interface in a multiple network interface device, after the first network interface sends an identifier, the identifier associated with a memory location in the multiple network interface device to a second device, the identifier and an associated data field capable of being received by the second network interface in the multiple network interface device; the acts comprising:

receiving a message from the second network interface by a program component in the multiple network interface device, the message indicating the reception of the identifier and the associated data field from the second device, wherein the second network interface has no knowledge of the identifier and the associated data field;

passing the identifier to the program component;

querying the first network interface to supply the program component with a list of identifiers generated by the first network interface and associated memory locations in multiple network interface device memory;

identifying, by the program component, that the first network interface generated the identifier; and

transmitting a memory location associated with the identifier to the second network interface, the second network interface capable of transmitting the associated data field to the memory location associated with the identifier.

On page 8, the Office Action alleges that Boyd in view of Applicant's admitted prior art "shows instructions stored on a computer readable medium (col. 35, lines 1-5 of Boyd) capable of performing the recited steps of claim 1." As should be clear from the above discussion in connection with claim 1, neither Boyd nor the Applicant's admitted prior art teach or suggest "instructions for performing acts for transferring control between a first network interface and at least a second network interface in a multiple network interface device, after the first network interface sends an identifier, the identifier associated with a memory location in the multiple network interface device to a second device, the identifier and an associated data field capable of being received by the second network interface in the multiple network interface device; the acts comprising: receiving a message from the second network interface by a program component in the multiple network interface device, the message indicating the reception of the identifier and the associated data field from the second device, wherein the second network interface has no knowledge of the identifier and the associated data field; passing the identifier to the program

component; querying the first network interface to supply the program component with a list of identifiers generated by the first network interface and associated memory locations in multiple network interface device memory; identifying, by the program component, that the first network interface generated the identifier; and transmitting a memory location associated with the identifier to the second network interface, the second network interface capable of transmitting the associated data field to the memory location associated with the identifier,” as recited in claim 15.

In view of the foregoing, claim 15 patentably distinguishes over Boyd and the Applicant's admitted prior art, either alone or in combination.

Claims 16-21 depend from claim 15 and are allowable for at least the same reasons.

Accordingly, withdrawal of the rejection of claims 15-21 is respectfully requested.

#### Rejections Under 35 U.S.C. §103     II

The Office Action rejected claims 2-3, 8-14, 16-17, 22-28 under 35 U.S.C. 103(a) as allegedly being unpatentable over Boyd in view of Applicant's admitted prior art and in further view of the Internet Draft document “RDMA Protocol Verbs Specification” by Jeff Hilland (“Hilland”). Applicants respectfully disagree.

#### C. Independent Claim 8

Claim 8 recites:

A method for transferring control between a first network interface and at least a second network interface in a host computer including the first network interface and the second network interface, the method comprising:

receiving an identifier from a remote computer by the at least a second network interface, the identifier generated by the first network interface and associated with a memory location in the host computer, wherein the second network interface has no knowledge of the identifier and the associated data field;

sending a message to a program component indicating the reception of the identifier, the program component configured to query the first network interface for a list of identifiers generated by the first network interface and associated memory locations in the host computer;

passing the identifier received from the remote computer to the program component;

searching the list of identifiers for the identifier;

if the list of identifiers includes the identifier received from the remote computer, receiving a memory location associated with the identifier; and

if the list of identifiers does not include the identifier received from the remote computer, invalidating the identifier received from the remote computer.

On page 10, the Office Action concedes that Boyd “does not show that the second network interface has no knowledge of the identifier and the associated data field, and that the first network interface generated the identifier.” The Office Action also concedes that Boyd “does not show searching the list of identifiers for the identifier, and, based on the result from the searching step, invalidating the identifier received from the remote computer if the list of identifiers does not include the identifier received from the remote computer.”

The Office Action then alleges that Applicant’s admitted prior art shows that the second network interface has no knowledge of the identifier and the associated data field, and that the first network interface generated the identifier. As discussed above in connection with claim 1, a combination of Boyd and the Applicant’s admitted prior art does not render claim 8 obvious because Boyd discusses sharing memory regions between Primary RNIC and the Alternate RNIC.

Further, the Office Action alleges that Hilland “shows searching the list of identifiers for the identifiers [using Stag by RNIC Interface (RI) when handling RDMA operations to identify a memory location within a specific Memory Regions (page 89, lines 42-28) wherein a memory regions is associated with a Physical Buffer List through the Stag] (page 13, lines 51).” Hilland does discuss that “for local access, the STag, along with a Tagged Offset (TO) is used by the RI, when processing a Work Requests SGE, to identify a memory location within a specific Memory Region.” (Hilland, page 88, Section 7.2.). However, in this portion, Hilland does not teach *searching* the list of identifiers for the identifier, as recited in claim 8. (Emphasis added).

Hilland also defines a Memory Region (MR) as an area of memory that the Consumer wants the RNIC to be able to (locally or locally and remotely) access directly in a logically contiguous fashion. A Memory Region is identified by an STag, a Base TO, and a length. A Memory Region is associated with a Physical Buffer List (PBL) through the Stag. (Hilland, page 11). Further, Hilland discusses that there are two types of Memory Regions. These are Non-Shared MR and Shared MR. *A Non-Shared MR has a PBL that is not shared with other MRs. A Shared MR has a PBL that may be shared with other MRs.* (Hilland, page 86). Thus, Hilland teaches that a Memory Region is identified by an Stag and the Memory Region has a Physical Buffer List. Therefore, while it is not clear to which identifier the Office Action refers to, in this portion, Hilland nevertheless does not describe **a list** of identifiers. (Emphasis added). Hilland



discusses the Physical Buffer List which is different from a list of identifiers. Thus, Hilland does not teach or suggest searching *the list of identifiers* for the identifier, as recited in claim 8. (Emphasis added).

The Office Action states that Hilland “also shows invalidating an identifier [Stag, to prevent RNIC from accessing memory locations via the Stag associated with that Memory Region].” However, nowhere does Hilland teach or suggest *if the list of identifiers does not include the identifier received from the remote computer*, invalidating the identifier received from the remote computer, as recited in claim 8.

In view of the foregoing, claim 8 patentably distinguishes over Boyd, the Applicant's admitted prior art and Hilland, either alone or in combination.

Claims 9-14 depend from claim 8 and are allowable for at least the same reasons.

Accordingly, withdrawal of the rejection of claims 8-14 is respectfully requested.

D. Independent Claim 22

Claim 22 recites:

A computer readable medium having stored therein instructions for performing acts for transferring control between a first network interface and at least a second network interface in a host computer including the first network interface and the second network interface, the method comprising:

receiving an identifier from a remote computer by the at least a second network interface, the identifier generated by the first network interface and associated with a memory location in the host computer, wherein the second network interface has no knowledge of the identifier and the associated data field;

sending a message to a program component indicating the reception of the identifier, the program component configured to query the first network interface for a list of identifiers generated by the first network interface and associated memory locations in the host computer;

passing the identifier received from the remote computer to the program component;

searching the list of identifiers for the identifier;

if the list of identifiers includes the identifier received from the remote computer, receiving a memory location associated with the identifier; and

if the list of identifiers does not include the identifier received from the remote computer, invalidating the identifier received from the remote computer.

On page 13, the Office Action rejects claim 22 using rejections similar to those applied to reject claim 8. As should be clear from the above discussion in connection with claims 1 and 8,

Boyd does not teach or suggest “receiving an identifier from a remote computer by the at least a second network interface, the identifier generated by the first network interface and associated with a memory location in the host computer, wherein the second network interface has no knowledge of the identifier and the associated data field; sending a message to a program component indicating the reception of the identifier, the program component configured to query the first network interface for a list of identifiers generated by the first network interface and associated memory locations in the host computer; passing the identifier received from the remote computer to the program component; searching the list of identifiers for the identifier; if the list of identifiers includes the identifier received from the remote computer, receiving a memory location associated with the identifier; and if the list of identifiers does not include the identifier received from the remote computer, invalidating the identifier received from the remote computer,” as recited in claim 22.

In view of the foregoing, claim 22 patentably distinguishes over Boyd, the Applicant's admitted prior art and Hilland, either alone or in combination.

Claims 23-28 depend from claim 22 and are allowable for at least the same reasons.

Accordingly, withdrawal of the rejection of claims 22-28 is respectfully requested.

**CONCLUSION**

A Notice of Allowance is respectfully requested. The Examiner is requested to call the undersigned at the telephone number listed below if this communication does not place the case in condition for allowance.

If this response is not considered timely filed and if a request for an extension of time is otherwise absent, Applicant hereby requests any necessary extension of time. If there is a fee occasioned by this response, including an extension fee, that is not covered by an enclosed check, please charge any deficiency to Deposit Account No. 23/2825.

Dated: March 5 2008

Respectfully submitted,

By: /James H. Morris/  
James H. Morris, Reg. No. 34,681  
Wolf, Greenfield & Sacks, P.C.  
Federal Reserve Plaza  
600 Atlantic Avenue  
Boston, Massachusetts 02210-2206  
Telephone: (617) 646-8000